PREMIUM AUDIT

NAME OF PROJECT	BartMAN
TOKEN BEP20 ADDRESS	0xea0af40425a8a7f7fcf965138bf11e3f8e12df7b
WEBSITE / TELEGRAM	http://BartMAN.io

DISCLAIMER: PLEASE READ FULL AUDIT

IMPORTANT DISCLAIMER

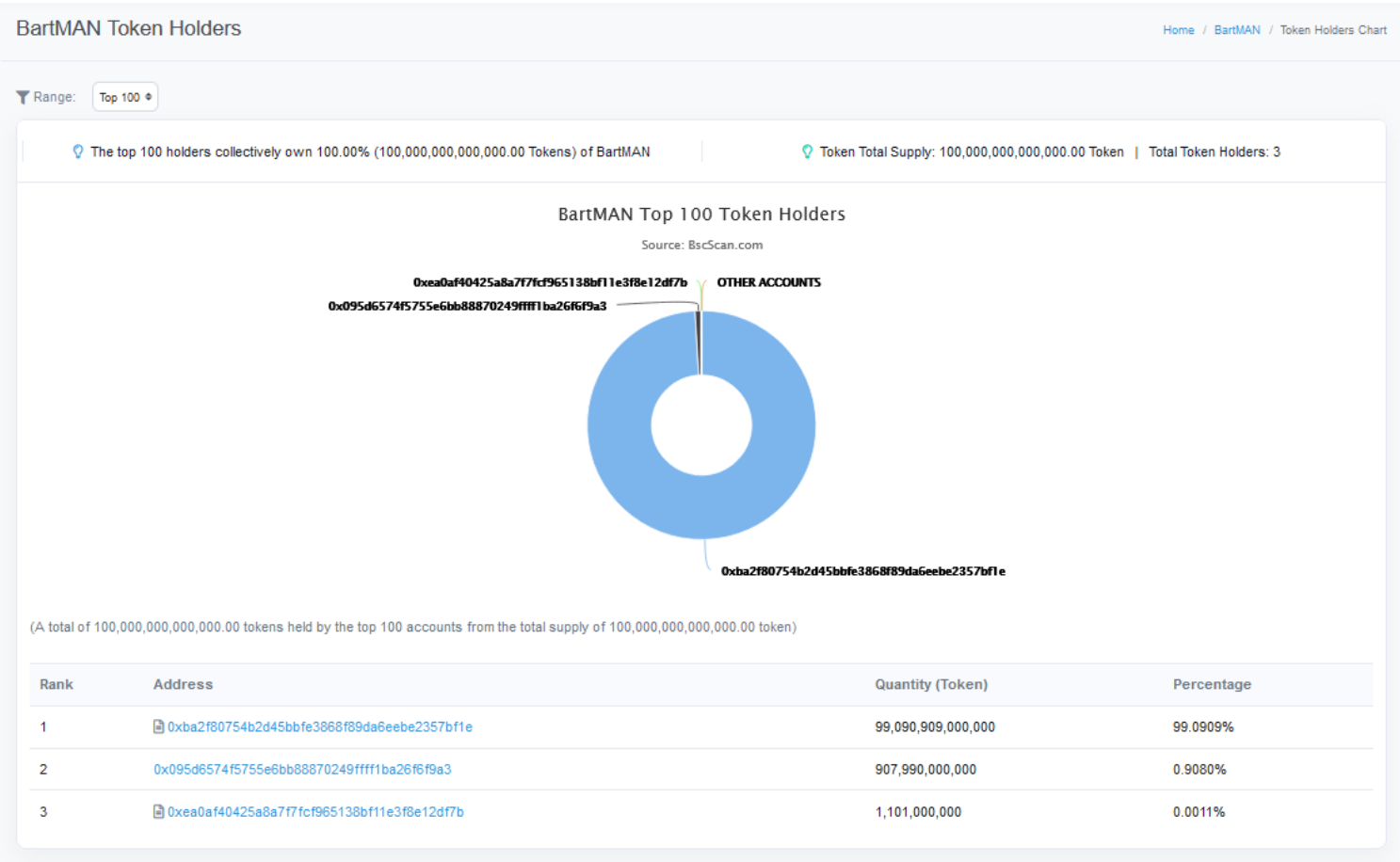|		
I.	**NOT RESPONSIBLE**	Analytix Audit holds no responsibility for any actions from the project in this audit.
II.	**NOT GUARANTEE**	Analytix Audit in no way guarantees that a project will not remove liquidity, sell off team tokens, or exit scam.
III.	**INFORMATION**	Analytix Audit researches and provides public information about the project in an easy-to-understand format for the common person.
IV.	**AUDIT AGREEMENT**	This audit agreement does not guarantee ANY illicit actions by the project team and does not serve as an advocation for the project.
V.	**NO ADVICE**	Analytix Audit in no way takes responsibility for any losses, not does Analytix Audit encourage any speculative investments.
VI.	**DISCLAIMER**	The information provided in this audit is for information purposes only and should not be considered investment advice. Analytix Audit does not endorse, recommend, support, or suggest any projects that have been audited.

TOKENOMICS - OVERVIEW

BartMAN Token Holders

Range: Top 100 ◇

The top 100 holders collectively own 100.00% (100,000,000,000,000.00 Tokens) of BartMAN | Token Total Supply: 100,000,000,000,000.00 Token | Total Token Holders: 3

BartMAN Top 100 Token Holders

Source: BscScan.com

0xea0af40425a8a7f7fcf965138bf11e3f8e12df7b — OTHER ACCOUNTS

0x095d6574f5755e6bb88870249ffff1ba26f6f9a3

0xba2f80754b2d45bbfe3868f89da6eebe2357bf1e

(A total of 100,000,000,000,000.00 tokens held by the top 100 accounts from the total supply of 100,000,000,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	0xba2f80754b2d45bbfe3868f89da6eebe2357bf1e	99,090,909,000,000	99.0909%
2	0x095d6574f5755e6bb88870249ffff1ba26f6f9a3	907,990,000,000	0.9080%
3	0xea0af40425a8a7f7fcf965138bf11e3f8e12df7b	1,101,000,000	0.0011%

*More than 15% tokens are unlocked

(This comment refers to when the audit was carried out maybe team locked tokens later)

CONTRACT CODE - PRIVILEGES

- **Mint:** Owner can't mint new tokens

- **Fees:** Owner can't change fees over 25%

- **Trading:** Owner can't change trading status

- **Max Tx:** Owner can't change maxTx

- **Max Wallet:** Owner can't change max wallet

- **Blacklist:** Owner can't blacklist wallet

Authorized Privileges:

```
519
520        * Requirements:
521        *
522        * - the calling contract must have an ETH balance of at least `value`.
523        * - the called Solidity function must be `payable`.
524        *
525        * _Available since v3.1._
526        */
527       function functionCallWithValue(
528           address target,
529           bytes memory data,
530           uint256 value
531       ) internal returns (bytes memory) {
532           return functionCallWithValue(target, data, value, "Address: low-level call with value failed");
533       }
534
535       /**
536        * @dev Same as {xref-Address-functionCallWithValue-address-bytes-uint256-}[`functionCallWithValue`], but
537        * with `errorMessage` as a fallback revert reason when `target` reverts.
538        *
539        * _Available since v3.1._
540        */
541       function functionCallWithValue(
542           address target,
543           bytes memory data,
544           uint256 value,
545           string memory errorMessage
546       ) internal returns (bytes memory) {
547           require(address(this).balance >= value, "Address: insufficient balance for call");
548           require(isContract(target), "Address: call to non-contract");
549
550           (bool success, bytes memory returndata) = target.call{value: value}(data);
```

function authorize(address adr) public onlyOwner {
function unauthorize(address adr) public onlyOwner {
function transferOwnership(address payable adr) public onlyOwner {
function setIsFeeExempt(address holder, bool exempt) external authorized {
function setBuyFees(uint256 _buyLiquidityFee, uint256 _buyMarketingFee)
function setSellFees(uint256 _SellLiquidityFee, uint256 _SellMarketingFee)
function setSwapBackSettings(bool _enabled, uint256 _amount)
function setTargetLiquidity(uint256 _target, uint256 _denominator)

*Owner itself cannot renounce ownership. - Can be recovered if renounced.

Issue Checking:

N°	Issue description.	Checking Status.
1	Compiler Errors	Passed
2	Race conditions and Reentrancy. Cross-function race conditions.	Passed
3	Possible delays in data delivery.	Passed
4	Oracle calls.	Passed
5	Front running.	Passed
6	Timestamp dependence.	Passed
7	Integer Overflow and Underflow.	Passed
8	DoS with Revert.	Passed
9	DoS with block gas limit.	Passed
10	Methods execution permissions.	Passed
11	Economy model.	Passed
12	The impact of the exchange rate on the logic.	Passed
13	Private user data leaks.	Passed
14	Malicious Event log	Passed
15	Scoping and Declarations.	Passed
16	Uninitialized storage pointers.	Passed
17	Arithmetic accuracy.	Passed
18	Design Logic.	Passed
19	Cross-function race conditions.	Passed
20	Safe Zeppelin module	Passed
21	Fallback function security.	Passed

Audit Result:

PASSED

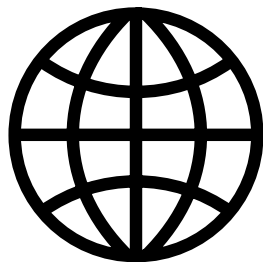# WEBSITE - OVERVIEW

SSL CERTIFICATE: A

(https://www.ssllabs.com)

SOCIAL MEDIA - OVERVIEW

BARTMAN_INU

BARTMAN_INU

BARTMAN.IO